Fault–Tolerant Minimum Broadcast Networks

R. Ahlswede
Fakultät für Mathematik
Universität Bielefeld
33501 Bielefeld 1
Postfach 100131, Germany

L. Gargano*
Dipartimento di Informatica ed Applicazioni
Universitá di Salerno
84081 Baronissi (Sa), Italy

H.S. Haroutunian
Fakultät für Mathematik
Universität Bielefeld
33501 Bielefeld 1
Postfach 100131, Germany

L.H. Khachatrian
Fakultät für Mathematik
Universität Bielefeld
33501 Bielefeld 1
Postfach 100131, Germany

Abstract

Broadcasting is the task of transmitting a message originated at one processor of a communication network to all other processors in the network. A minimal k–fault tolerant broadcast network is a communication network on n vertices in which any processor can broadcast in spite of up to k line failures in optimal time $T_n(k)$. In this paper we study $B_k(n)$, the minimum number of communication lines of any minimal k–fault tolerant broadcast network on n processors. We give the value of $B_k(n)$ for several values of n and k and, in case $k < \lfloor \log n \rfloor$, give almost–minimum k-fault–tolerant broadcast networks.

*Research partially done while visiting the Sonderforschungsbreich "Diskrete Strukturen in der Mathematik" of Bielefeld University. The work of this author was partially supported by the Italian Ministry of University and of Scientific Research in the framework of the "Algoritmi, Modelli di Calcolo e Strutture Informative" project.

1 Introduction

This paper deals with graphs in which broadcasting can be performed efficiently even in presence of failures.

We represent a communication network as a connected undirected graph $G = (V, E)$ where the node set V represents the set of processors and the edge set represents the set of bidirectional communication lines between processors. *Broadcasting* is the process of delivering a message from a processor (called the *originator*) to all the other processors in the network. The communication model considered in this papers assumes the following constraints:

(1) Messages can be sent directly only to neighbors in the graph;

(2) each message transmission (*call*) requires one unit of time;

(3) any processor can participate in at most one call per time unit.

The broadcast problem in this model has been studied by several authors [4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 22, 19, 20, 25, 26, 27, 29, 30]. Recently, in [1] new concepts of broadcasting have been introduced.

Two important measures of the goodness of a network are the time needed to perform the broadcast and the number of communication lines.
It is easy to see that after t time units the maximum number of members which may have received the message (*informed nodes*) is 2^t, including the originator. This is due to the fact that, in one unit of time, the number of informed nodes can at most double. Furthermore, this implies that the minimum number of time units required to broadcast a message in a set of n members is $\lceil \log n \rceil$[1].
A *minimal broadcast network (mbn)* is a communication network with n nodes in which a message can be broadcasted in $\lceil \log n \rceil$ time units, regardless of the originator.
Let $B(n)$ be the minimum number of edges in any mbn with n nodes. Minimal broadcast networks having $B(n)$ edges are called *minimum broadcast networks*; they represent the cheapest possible broadcast networks, as far as the number of communication lines is concerned, in which the broadcast from any node requires minimum time $\lceil \log n \rceil$. Values of $B(n)$ for some n are known; however, no general method of constructing mbn's is known. The problem of determining $B(n)$ for arbitrary n is thought to be NP-complete. Notice that the problem of recognizing whether a given network is a minimal broadcast network is NP-complete. Since the design of minimum broadcast networks with n nodes, n arbitrary, seems to be very difficult, much attention has been given to the construction of mbn's with as few edges as possible [9, 4, 15, 18, 7, 22, 19, 20].

[1]All logarithms in this paper are in base 2.

Liestman [25] first introduced the important issue of fault–tolerant broadcasting. Consider a network $G = (V, E)$ and let $E' \subset E$ be any set of edges, $|E'| \leq k$. The set of edges E' represents faulty communication lines and the subgraph $G' = (V, E - E')$ represents the functioning part of the network. A k *fault–tolerant broadcasting* (k–ftb) scheme is a broadcast protocol which assures that any node in the network will receive the message from the originator in presence of up to k line failures, i.e., when *any* set E' of size $|E'| \leq k$ is faulty. It is important to point out that in this model the sequence of calls is fixed and cannot be changed when faults are detected.

Indicate by $T_k(n)$ the minimum integer such that there exists a network on n nodes in which any node can perform a k-fault tolerant broadcast in at most $T_k(n)$ time units.

The best known lower bounds on $T_k(n)$ are $\lceil \log n \rceil + k$ for general values of k and n and $T_k(n) \geq \lceil \log n \rceil + k + 1$ for each k and odd n such that $k \geq 2^{\lceil \log n \rceil} - n + \lfloor \log(2^{\lceil \log n \rceil} - n - 1) \rfloor$, [25, 16, 28]. In [5] and [25] k–ftb schemes requiring optimal time were given for $k = 1$ and 2. Lately, it was shown in [16] that for each $n \geq 8$

$$T_k(n) = \begin{cases} \lceil \log n \rceil + k & \text{for } n \text{ even and } k \leq \lfloor \log n \rfloor \text{ or} \\ & \quad n \text{ odd and } k \leq \lfloor \log(2^{\lceil \log n \rceil} - n + 1) \rfloor; \\ \lceil \log(n-1) \rceil + k + 1 & \text{for } n \text{ odd and } k \text{ satisfying the two inequalities} \\ & \quad k \geq 2^{\lceil \log n \rceil} - n - 1 + \lfloor \log(2^{\lceil \log n \rceil} - n + 1) \rfloor \\ & \quad k \leq \lfloor \log(n-1) \rfloor \\ \lceil \log n \rceil + k \text{ or } \lceil \log(n-1) \rceil + k + 1 & \text{for } n \text{ odd and the remaining } k \leq \lfloor \log(n-1) \rfloor. \end{cases}$$

The problem of broadcasting in the presence of a larger number of failures has been considered in [26], [28], and [13]. In [26] and [28] it is shown that $T_k(n) \leq \lceil \log \lceil \frac{n}{k+1} \rceil \rceil + 2k$. Moreover, in [28] Peleg and Schäffer show that $T_k(n) \leq 2c\lceil \log n \rceil + k + 8k/9c + d$, where $c \geq 1$ is an integer such that $k \leq (n-1)/(8c+1) - c$, and d is a constant.

In [13] it is shown that that if $n \leq 2^{\lfloor \log n \rfloor} + 2^{\lfloor \log n \rfloor - 2}$ then

$$T_k(n) \leq \begin{cases} \lceil \log n \rceil + k + \left\lfloor \frac{k}{\lceil \log n \rceil - \lceil \log(n - 2^{\lfloor \log n \rfloor}) \rceil - 2} \right\rfloor & \text{if } k \leq 2^{\lfloor \log n \rfloor} - \lceil \log n \rceil \\ \lceil \log n \rceil + k + \left\lfloor \frac{k-1}{\lceil \log n \rceil - \lceil \log(n - 2^{\lfloor \log n \rfloor}) \rceil - 2} \right\rfloor + 1 & \text{if } 2^{\lfloor \log n \rfloor} - \lceil \log n \rceil < k \leq 2^{\lfloor \log n \rfloor} - 2. \end{cases}$$

and that

$$T_k(2^m) = \begin{cases} m + k & \text{if } k \leq 2^m - m - 1 \\ m + k + 1 & \text{if } 2^m - m \leq k \leq 2^m - 2 \end{cases}.$$

In this paper we consider the problem of constructing sparse broadcast networks supporting k fault–tolerant broadcasting schemes requiring minimum time.

Call *minimal* k-fault–tolerant broadcast network a graph in which any node can perform k fault–tolerant broadcasting in time $T_k(n)$.

Let $B_k(n)$ be the minimum number of edges in any minimal k-fault–tolerant broadcast network with n nodes. k–fault–tolerant broadcast networks having $B_k(n)$ edges represent the cheapest possible broadcast networks, as far as the number of communication lines is concerned, in which fault–tolerant broadcasting from any node requires the minimum possible time for the given values of n and k. They are called *minimum* k–fault–tolerant broadcast networks.

The problem of evaluating $B_k(n)$ has been studied in [4] and [25] in case $k = 1$ and 2. Gargano and Vaccaro [16] consider $k < \lfloor \log n \rfloor$ and show that $B_k(n) = O(n(k + \lceil \log n \rceil)/2)$. Moreover, Gargano [13] shows that $B_k(2^m) = (m+k)2^{m-1}$ for each $k \leq 2^m - m$. In [20] L. Khachatrian and H. Haroutiounian show that $B_1(2^m - 2) = m(2^{m-1} - 1)$.

In this paper we study the construction of minimum or almost–minimum k-fault–tolerant broadcast networks. In particular we propose k-fault–tolerant broadcast networks whose number of edges differs from the minimum possible number of edges of any k-fault–tolerant broadcast network in at most a constant factor, for $k < \lfloor \log n \rfloor$ if n is even and $k \leq \lfloor 2^{\lceil \log n \rceil} - n + 1 \rfloor$ if n is odd.
Moreover, we derive the exact value of $B_k(n)$ for several values of n and k and give graphs having $B_k(n)$ edges. In particular for $k = 1$ we extend the work in [5, 25] by giving $B_1(n)$, for each $n \leq 16$, and $n = 2^m - 6$, for any m.

2 A lower bound

In this section we establish a lower bound on $B_k(n)$ for n and k such that $T_k(n) = \lceil \log n \rceil + k$.

Let the number of nodes be n such that $n - 1 = 2^m - 2^\ell + j$ with $m = \lceil \log n - 1 \rceil$, $\ell < m$, and $0 \leq j < 2^{\ell-1}$. Define

$$L(n) = \begin{cases} m - \ell + 1 & \text{if } j > 0 \\ m - \ell & \text{if } j = 0. \end{cases}$$

Lemma 2.1 *Any vertex v in a minimal k–fault–tolerant broadcast networks G on n vertices must have degree* $\deg(v) \geq k + L(n)$.

Proof. Let the vertex v be the originator and v_i, for $i = 1, 2 \ldots,$, the i–th neighbor called by v. Since any set of k edges can be faulty, let us suppose that the edges $(v, v_1), (v, v_2), \ldots, (v, v_k)$ are the faulty ones. Vertex v_{k+i} is called at time unit $t_i \geq k+i$ and in the remaining $(\lceil \log n \rceil + k) - t_i \leq \lceil \log n \rceil - i$ time units at most $2^{\lceil \log n \rceil - i} - 1$ other nodes can be informed through v_{k+i}. Since $n - 1$ nodes must receive the information v must call at least $v_{k+1}, \ldots, v_{k+t}$ where

$$\sum_{i=1}^{t-1} 2^{\lceil \log n \rceil - i} = 2^{\lceil \log n \rceil} - 2^{\lceil \log n \rceil - t + 1} < n - 1 \leq \sum_{i=1}^{t} 2^{\lceil \log n \rceil - i} = 2^{\lceil \log n \rceil} - 2^{\lceil \log n \rceil - t}.$$

Therefore, $t = L(n)$ and $\deg(v) \geq k + L(n)$. □

We then get the following theorem.

Theorem 2.1 *For each n and k such that $T_k(n) = \lceil \log n \rceil + k$ it holds $B_k(n) \geq (k + L(n))n/2$.*

3 A simple case: $n = 2^m$

In this section we summarize the case $n = 2^m$ and $k \leq n - m$ solved in [13].

We describe the calling scheme and give a graph which supports it. In order to illustrate the calling scheme let us introduce some notation. Label, in any order, the nodes with the different elements of F_{2^m}, the finite field with 2^m elements (For notions on finite fields see [24]). Let α be a primitive element of the field F_{2^m}, that is, α is such that its successive powers $\alpha^0 = 1, \alpha^1, \ldots, \alpha^{n-2}$ give all the nonzero elements of the field and $\alpha^{n-1} = 1$.

Broadcast–Scheme(F_{2^m}):
Time unit t, ($t = 1, \ldots, m + k \leq n - 1$): each informed node β calls $\beta + \alpha^{t-1}$.

Theorem 3.1 *[13] For each $k \leq n - m - 1$ the graph $G = (V, E)$ with $V = F_{2^m}$ and $E = \{(\beta, \beta + \alpha^i) | \alpha$ is a primitive element of F_{2^m} and $i = 0, \ldots, m + k - 1\}$ allows a k-fault tolerant broadcast in minimum time and has the minimum possible number of edges.*

Corollary 3.1 $B_k(2^m) = (m + k)2^{m-1}$ *for each* $k \leq 2^m - m$.

4 Fault–Tolerant Broadcast Graphs on an even number of nodes

In this section we give near–minimum k–fault tolerant graphs for n even and $k < \lfloor \log n \rfloor$.

Let the node set be $\{0, \ldots, n-1\}$, with n even and $n \neq 2^{\lceil \log n \rceil}$, and let $k < \lfloor \log n \rfloor$. Define the sets

$$V_t = \left\{ \left\lceil \frac{n}{2^t} \right\rceil, \left\lceil \frac{n}{2^t} \right\rceil + 1, \ldots, \left\lceil \frac{n}{2^{t-1}} \right\rceil - 1 \right\} \qquad \text{for } t = 1, \ldots, k \tag{1}$$

Note that the sizes of the sets V_t satisfy the following relations:

$$|V_t| \in \left\{ \lceil n/2^t \rceil, \lceil n/2^t \rceil - 1 \right\} \tag{2}$$

and

$$\lceil n/2^t \rceil = n - \sum_{i=1}^{t} |V_i|. \tag{3}$$

Let the set

$$B = \{0, \ldots, n-1\} \setminus \bigcup_{t=1}^{k} V_t = \left\{ 0, \ldots, \left\lceil \frac{n}{2^k} \right\rceil - 1 \right\} \tag{4}$$

with

$$\lceil \log |B| \rceil = \lceil \log n \rceil - k. \tag{5}$$

If $\left\lceil \frac{n}{2^k} \right\rceil = |B| < 2^{\lceil \log n \rceil - k}$ we can define ℓ and j such that ℓ is the minimum integer for which

$$|B| = 2^{\lceil \log n \rceil - k} - 2^{\ell} + j, \qquad \text{with } \ell < \lceil \log n \rceil - k, \ \text{ and } j < 2^{\ell - 1}; \tag{6}$$

if $\left\lceil \frac{n}{2^k} \right\rceil = |B| = 2^{\lceil \log n \rceil - k}$ we (formally) put $\ell = 0$ and $j = 1$.

For each integer $t > k$ define the sets V_t as follows

$$V_t = \left\{ j + \sum_{i=\ell}^{\lceil \log n \rceil - t - 1} 2^i, \ \ldots, \ j + \sum_{i=\ell}^{\lceil \log n \rceil - t} 2^i - 1 \right\} \quad \text{for } t = k+1, \ldots, \lceil \log n \rceil - \ell - 1 \tag{7}$$

$$V_{\lceil \log n \rceil - \ell} = \{ j, j+1 \ldots, j + 2^{\ell} - 1 \} \tag{8}$$

$$V_{\lceil \log n \rceil - \ell + 1} = \{ 0, \ldots, j - 1 \} \tag{9}$$

$$|V_t| = 2^{\lceil \log n \rceil - t}, \qquad \text{for } t = k+1, \ldots, \lceil \log n \rceil - \ell, \qquad |V_{\lceil \log n \rceil - \ell + 1}| = j \tag{10}$$

Indicate by v_t node with smallest (highest if $t = 1$) label in the set V_t (as defined in (1) and (7)), that is,

$$v_t = \begin{cases} \max\{v \ : \ v \in V_1\} = n - 1 & \text{if } t = 1 \\ \min\{v \ : \ v \in V_t\} = \lceil \frac{n}{2^t} \rceil & \text{if } 2 \le t \le k; \\ \min\{v \ : \ v \in V_t\} = j + \sum_{i=\ell}^{\lceil \log n \rceil - t - 1} 2^i & \text{if } k+1 \le t \le \lceil \log n \rceil - \ell - 1. \\ \min\{v \ : \ v \in V_t\} = j(\lceil \log n \rceil - \ell - t + 1) & \text{if } \lceil \log n \rceil - \ell \le t \le \lceil \log n \rceil - \ell + 1 \end{cases} \tag{11}$$

4.1 Broadcast trees

In this section we describe a broadcast tree on a set of n nodes labelled by consecutive integers.

The broadcast process proceeds as follows. Initially, at time unit 0, only the originator knows the message. In general after time unit $t - 1$ there are 2^{t-1} informed nodes and each of them has to broadcast on a subset of consecutive numbered nodes. This set is divided into two subsets of consecutive numbered nodes such that either they have equal size or the one the informed node belongs to has one element more than the other. The informed node calls a node in the other subset, that will broadcast on it, while it continues broadcasting on its subset. The process continues for $\lceil \log n \rceil$ time units at the end of which there is a calling path from the originator to each node in the network.

The broadcast tree on the set of all nodes consecutively numbered from a to b and rooted in the originator v is the tree

$$T(\{a, \ldots, b\}, v) = (\{a, a+1, \ldots, b\}, E)$$

where the edge set E is obtained by inserting the edge (u, w) if and only if the above described broadcasting process poses a call from u to w.

Here we specify, for further use, a slightly different calling process (that is, edges in $T(V_1, v_1)$). when the node set is $V_1 = \{n/2, \ldots, n-1\}$ and the root is the node with label $v_1 = n-1$. Consider the partition of the node set $\{n/2, \ldots, n-1\}$ defined, following (1) and (7), by

$$A'_t = \{n-1-v \ : \ v \in V_{t+1}\} \quad \text{for } t = 1, \ldots, \lceil \log n \rceil - \ell - 1 \tag{12}$$

$$A'_{\lceil \log n \rceil - \ell} = \{n-1-v \ : \ v \in V_{\lceil \log n \rceil - \ell + 1}, \ v \neq n-1\}. \tag{13}$$

Consider then the following broadcast scheme from $n-1$ to $\{n/2, \ldots, n-1\}$:

> **time unit** t $\quad (1 \leq t \leq \lceil \log n \rceil - \ell)$**:** $n-1$ calls a node in A'_t that, during the following $\lceil \log n \rceil - 1 - t$ time units, broadcasts to the $|A'_t| \leq 2^{\lceil \log n \rceil - 1 - t}$ nodes in A'_t.

The broadcast from the informed node in A'_t to the other nodes in A'_t can be performed in $\lceil \log n \rceil - 1 - t$ time units according to the protocol previously outlined. Therefore, the above broadcasting scheme informs each node in $\{n/2, \ldots, n-1\}$ in $\lceil \log n \rceil - 1$ time units.

Example 4.1 *Let the node set be* $\{0, \ldots 19\}$ *and* $k = 2$*. One has* $\lceil 19/4 \rceil = 5$ *with* $\ell = 2$ *and*

$$\begin{array}{ll} V_1 = \{10, \ldots, 19\} & \\ V_2 = \{5, 6, 7, 8, 9\} & A'_1 = \{14, 13, 12, 11, 10\} \\ V_3 = \{1, 2, 3, 4\} & A'_2 = \{18, 17, 16, 15\} \\ V_4 = \{0\} & A'_3 = \{19\} \end{array}$$

The trees $T(V_2, 5)$ *and* $T(V_1, 19)$ *are represented in Figure 1.*

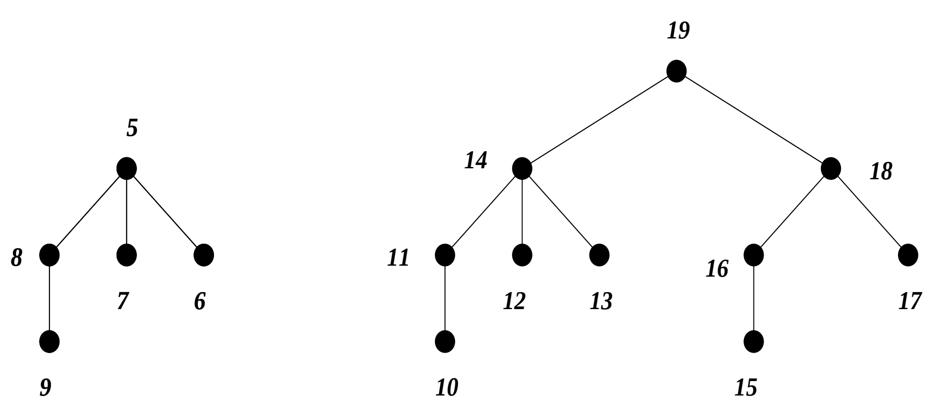


Figure 1:

4.2 The Graphs

Let $T_t = T(V_t, v_t)$ the broadcast tree (as defined in the previous section) on the node set V_t rooted in v_t.

Denote by E_t the edge set of the tree T_t, for $t = 1, \ldots, \lceil \log n \rceil - \ell + 1$. Moreover, let

- $E^{(1)} = \bigcup_{t=1}^{\lceil \log n \rceil - \ell + 1} E_t$;
- $E^{(2)}$ be the following set of edges: for each t and each v node in T_t if v is a son of f and has sons $s_1, \ldots, s_r$ (called in this order during the broadcast on T_t from its root v_t), that is, $(f, v), (v, s_1), \ldots, (v, s_r) \in E_t$, then $E^{(2)}$ contains the edges $(f, s_1), (s_1, s_2) \ldots, (s_{r-1}, s_r)$; if $v = v_t$, the root of T_t, and has sons $s_1, \ldots, s_r$ then $E^{(2)}$ contains the edges $(s_1, s_2) \ldots, (s_{r-1}, s_r)$;
- $E^{(3)}$ be the following set of edges: connect to each v_t each node which is not already connected to v_t. That is, $E^{(3)}$ contains all the edges (v, v_t) such that $(v_t, v) \notin E_t \cup E^{(2)}$.
- $E^{(4)}$ be the following set of edges: for each node $v \in \{0, \ldots n/2 - 1\}$ and $t = 1, \ldots, k$ define

  $$f_t^{(n)}(v) = \begin{cases} \frac{n}{2} + (v \oplus \lceil \frac{n}{2^t} \rceil) & \text{if } v < n/2 \\ (v \ominus \lceil \frac{n}{2^t} \rceil) & \text{if } v \geq n/2 \end{cases}$$

  where $\ominus$ and $\oplus$ denote, respectively, the subtraction and the addition modulo $n/2$, then

  $$E^{(4)} = \left\{ (v, f_t^{(n)}(v)) \ : \ 0 \leq v \leq n - 1, \ 1 \leq t \leq k \right\}$$

Define then the graph

$$G_{k,n} = (\{0, \ldots, n-1\}, E_{k,n}), \qquad \text{where } E_{k,n} = E^{(1)} \cup E^{(2)} \cup E^{(3)} \cup E^{(4)}.$$

In the sequel we prove the following results.

Theorem 4.1 *The graph $G_{k,n}$ allows k–fault–tolerant broadcasting from each node in minimum time for each n even and $k \leq \lfloor \log n \rfloor$.*

Proof. We give a broadcasting scheme on $G_{n,k}$ and prove that it is k–fault–tolerant. Call x the originator. The process is in two rounds.

BROADCAST–SCHEME($G_{n,k}, x$) *[Suppose* $|B| = \lceil \frac{n}{2^k} \rceil \neq 2^{\lceil \log n \rceil - k}$*]*
ROUND 1:
(1) If $x = v_1$ then
(2) **time unit** 1**:** v_1 calls the node v_2
(3) **time unit** t $(1 \leq t \leq \lceil \log n \rceil - \ell)$**:** v_t calls the node v_{t+1} and during the following $\lceil \log n \rceil - t$ time units, broadcasts to the $|E_t| \leq 2^{\lceil \log n \rceil - t}$ nodes in T_t.
v_1 calls a node in A'_t (as defined in (1)) which during the following $\lceil \log n \rceil - t$ time units, broadcasts to the $|A'_t| \leq 2^{\lceil \log n \rceil - t}$ nodes in A'_t.
(4) If x is a node in T_i with $x \neq v_i$ then
(5) **time unit** t $(1 \leq t \leq \lceil \log n \rceil - \ell)$**:** x calls the node v_t that during the following $\lceil \log n \rceil - t$ time units, broadcasts to the $|E_t| \leq 2^{\lceil \log n \rceil - t}$ nodes in T_t;
(if $v_1 = f_\tau(x)$, for some $1 \leq \tau \leq k$, then at time 1 x calls the son that v_1 first informs when it broadcasts on T_1)
(6) If $x = v_i$ with $i \neq 1$ then
(7) **time unit** t $(1 \leq t \leq \lceil \log n \rceil - \ell,\ t \neq i)$**:** x calls the node v_t that during the following $\lceil \log n \rceil - t$ time units, broadcasts to the $|E_t| \leq 2^{\lceil \log n \rceil - t}$ nodes in T_t.
(8) **time unit** $t = i$**:** *[call* $s_1, \ldots, s_r$ *the sons of* x*]* x calls the node s_1,
s_1 will call s_2 at time unit $i+1$; s_2 will call s_3 at time unit $i+2$, and so on.
ROUND 2:
(9) **time unit** $\lceil \log n \rceil + t$ $(1 \leq t \leq k)$**:** each node v, if informed, calls the node $f_t^{(n)}(v)$.

Notice that in Round 1 of BROADCAST–SCHEME($G_{n,k}, x$), if the originator x is a node in T_t then the broadcast from v_t is modified as follows to avoid the node x: Let f be the father of x and $s_1, \ldots, s_r$ be its sons, during the broadcast on T_t f calls x at time i and x calls s_j at time $i+j$, then in the modified broadcasting f calls s_1 at time i and s_j calls s_{j+1} at time $i+j$ (these calls, as the calls of line (9), are possible since the edges of $G_{n,k}$ include those in $E^{(2)}$).

Consider now $|B| = \lceil \frac{n}{2^k} \rceil = 2^{\lceil \log n \rceil - k}$. The above reasoning implies that if the originator is a node in the tree T_i then there certainly is a node, say u, in this tree that ends Round 1 before the time units $\lceil \log n \rceil$. This implies that we can easily modify the above algorithm to handle the case $|B| = 2^{\lceil \log n \rceil - k}$ just letting u call $v_{\lceil \log n \rceil + 1} = 0$ at time $\lceil \log n \rceil$. The broadcast from $v_{\lceil \log n \rceil + 1}$ is the same as described for the other v_i's.

It is immediate to see that in absence of faults, each node receives the originator message within the $\lceil \log n \rceil$ time units of Round 1 of the BROADCAST–SCHEME($G_{n,k}, x$).
In particular, Round 1 creates a broadcast tree rooted in the originator x such that if $x \neq v_1$ the subtree rooted in the node called at time unit t has node set V_t for $t = 1, \ldots \lceil \log n \rceil - \ell + 1$; if $x = v_1$ the subtree rooted in the node called at time unit t has node set A'_t, for $t = 1, \ldots \lceil \log n \rceil$

We show now that Round 2 completes a k–fault–tolerant broadcasting, that is, we show that during each of the k time units of Round 2 each node receives the message of the originator along an additional edge disjoint path.

We call *informer* of a node v at time t a node a from which v receives the message (in the absence of failures) during the first t time units of Round 2 (that is, within time unit $\lceil \log n \rceil + t$ of the BROADCAST–SCHEME$(G_{n,k}, x)$). More precisely, a is an informer of v at time t if either $a = v$ or the calls of the first t time units of Round 2 form a path

$$(a, f_{i_1}(a) = a_1)(a_1, f_{i_2}(a_1) = a_2) \cdots (a_{r-1}, f_{i_r}(a_{r-1}) = v),$$

for some integers $i_1, i_2, \ldots, i_r$, $1 \le r \le t$, such that $1 \le i_1 < \ldots < i_r \le t$. We denote by $\inf(v,t)$ the set of informers of v at time t. Formally, $\inf(v, 0) = \{v\}$ and for each $t \le k$ we have

$$\inf(v,t) = \{v\} \cup \{a |\text{ there exist integers } 1 \le i_1 < \ldots < i_r \le t \text{ such that } f_{i_r}(\ldots(f_{i_1}(a))) = v\}.$$

$$\begin{aligned}\inf(v,t) &= \inf(v,t-1) \cup \{a |\text{ exist } 1 \le i_1 < \ldots < i_s \le t-1 \text{ with } f_t(f_{i_s}(\ldots(f_{i_1}(a)))) = v\} \\ &= \inf(v,t-1) \cup \{a |\text{ exist } 1 \le i_1 < \ldots < i_s \le t-1 \text{ with } f_{i_s}(\ldots(f_{i_1}(a))) = f_t(v)\} \cup \{f_t(v)\}\end{aligned}$$

and therefore

$$\inf(v,t) = \inf(v,t-1) \cup \inf(f_t(v), t-1) \tag{14}$$

Consider as example $n = 10$ and $v = 3$: one has

$$\inf(3,0) = \{3\}, \quad \inf(3,1) = \{3,8\}, \quad \inf(3,2) = \{1,3,6,8\}, \quad \inf(3,3) = \{0,1,2,3,5,6,7,8\}$$

Let S_t be the set formed by 0 and the $2^{t-1} - 1$ possible sums of elements in $\{|V_2|, \ldots, |V_t|\}$, e.g. $S_1 = \{0\}$, $S_2 = \{0, |V_2|\}$, $S_3 = \{0, |V_2|, |V_3|, |V_2| + |V_3|\}$, and so on. Note that $\{|V_2|, \ldots, |V_t|\} = \{|A'_1|, \ldots, |A'_{t-1}|\}$ The following properties of the sets $\inf(v,t)$ are proved in [16].

1) For each node v and time instant t, $1 \le t \le k$,

$$inf(v,t) = \begin{cases} \{v \ominus z | z \in S_t\} \cup \{n/2 + (v \ominus z) | z \in S_t\} & \text{if } v < n/2, \\ \\ \{v \oplus z | z \in S_t\} \cup \{n/2 + (v \oplus z) | z \in S_t\} & \text{if } v \ge n/2. \end{cases} \tag{15}$$

2) For each node v and time instant t, $1 \le t \le k$, the set of informers of v at time t, $\inf(v,t)$, is equal to a set $\{a_1, \ldots, a_{2^t}\}$ with $a_1 < a_2 < \cdots < a_{2^t}$ satisfying

$$n + (a_1 - a_{2^t}), a_i - a_{i-1} \in \{\lceil n/2^t \rceil, \lceil n/2^t \rceil - 1\}, \qquad i = 2, \ldots, 2^t. \tag{16}$$

3) For each node v and time instant t, $1 \le t \le k$,

$$\inf(v,t-1) \cap \inf(f_t(v), t-1) = \emptyset. \tag{17}$$

A *t–path to a node v* is any calling path from the originator to v formed by calls of Round 1 and of the first t time units of Round 1.
Given a t–path P to a node v, there exists exactly one node $I(P) \in \inf(v,t)$ such that $P = RQ$ where:

- R is a path, created by the calls of Round 1, from the originator to $I(P)$;
- Q is a path from $I(P)$ to v which is created by the calls of the first t time units of Round 2, that is $Q = (I(P), f_{i_1}(I(P)) = a_1)(a_1, f_{i_2}(a_1) = a_2)\ldots(a_{i_r-1}, f_{i_r}(a_{i_r-1}) = v)$ for some integers $i_1, i_2, \ldots, i_r$ such that $1 \le i_1 < \ldots < i_r \le t$; Q is empty if $I(P) = v$.

Notice that for each originator x the edges used in the execution of Round 1 are different from those used in Round 2. From (17), following the lines of the analogous result one presented in [16] we can prove the following property of the above paths: Given a $(t-1)$–path $P = RQ$ from the originator to node v and a $(t-1)$–path $P' = R'Q'$ from the originator to $f_t(v)$, the paths P and $P'(f_t(v), v)$ are edge disjoint if and only if R and R' are edge disjoint.
From the above property of t–paths and (16) it is easy to derive that for each originator x, the sequence of $\lceil \log n \rceil$ calls performed in Round 1 followed by the calls of the $k \le \lfloor \log n \rfloor$ time units of Round 2 give a k fault–tolerant broadcast protocol. Therefore, the graph $G_{n,k}$ allows k–fault–tolerant broadcasting in time $\lceil \log n \rceil + k$ from each node. $\square$

Theorem 4.1 gives the desired bound on $B_k(n)$ when n is even. Notice that $L\left(\lceil \frac{n}{2^k} \rceil + 1\right) \le L(n) + 1$ for each n and $k \le \lceil \log n \rceil$.

Corollary 4.1 *For each n even and $k \le \lfloor \log n \rfloor$*

$$B_k(n) \le |E_{k,n}| \le \left(\frac{3}{2}k + L\left(\lceil \frac{n}{2^k} \rceil + 1\right) + 2\right)(n-3) + \frac{3}{2}k + 6.$$

Proof. From the definition of the edge set $E_{n,k}$ of the graph $G_{n,k}$, we have $E_{n,k} = \sum_{i=1}^{4} |E^{(i)}|$ with

$$|E^{(1)}| = |\bigcup_{t=1}^{\lceil \log n \rceil - \ell + 1} E_t| \le n - \lceil \log n \rceil - \ell + 1 = k + L\left(\lceil \tfrac{n}{2^k} \rceil + 1\right);$$
$$|E^{(2)}| \le n - \lceil \log n \rceil - \ell + 1 = k + L(\lceil \tfrac{n}{2^k} \rceil + 1);$$
$$|E^{(3)}| \le (n-1)\left(k + L\left(\lceil \tfrac{n}{2^k} \rceil + 1\right)\right);$$
$$|E^{(4)}| = n\tfrac{k}{2}.$$

Hence $E_{n,k} = \sum_{i=1}^{4} |E^{(i)}| \le \left(\frac{3}{2}k + L\left(\lceil \frac{n}{2^k} \rceil + 1\right) + 2\right)(n-3) + \frac{3}{2}k + 6.$ $\square$

4.3 Odd number of nodes

Let the number n of nodes in the network be odd. We show that

Theorem 4.2 *For each $k \leq \lfloor 2^{\lceil \log n \rceil} - n + 1 \rfloor$ the graph*

$$H_{n,k} = (\{0, \ldots, n-1\}, E_{n-1,k} \cup \{(n-1, v) \ : \ 0 \leq v \leq n-2\}) \tag{18}$$

allows k–fault–tolerant broadcasting in time $\lceil \log n \rceil + k$ from each node.

Proof. Let the node set be $\{0, \ldots, n-1\}$ with n odd. The idea is to apply the BROADCAST–SCHEME($G_{n-1,k}, x$) to the even sized set of nodes $\{0, \ldots, n-2\}$ with the following modifications.

BROADCAST–SCHEME–ODD($G_{n,k}, x$)

ROUND 1:
If the originator is the node $n-1$ then
  time unit t $\quad (1 \leq t \leq \lceil \log n \rceil - \ell)$**:**
    the originator $n-1$ calls the root v_t which in the following time units will broadcast to nodes in T_t
else *[the originator is the node $x \neq n-1$]*
  time unit t $\quad (1 \leq t \leq \lceil \log n \rceil - \ell)$**:**
    Calls are the same as in Round 1 of BROADCAST–SCHEME($G_{n,k-1}, x$) but for the fact that,
    if the originator x is a node in the tree T_i then during the the broadcasting to the nodes in T_i
    node x is not eliminated but substituted by $n-1$.
    [Notice that the above substitution can be done since the considered values of k assure that
    $|B| = \lceil (n-1)/2^k \rceil < 2^{\lceil \log n-1 \rceil - k}$ (cfr. (4), (5)].

ROUND 2:
time unit $\lceil \log(n-1) \rceil + t$ $\quad (1 \leq t \leq k)$**:**
    each informed node $v \in \{0, \ldots, n-2\}$ with $v \neq x$ and $v \neq f_t^{(n-1)}(x)$, calls the node $f_t^{(n-1)}(v)$
    if $x \neq n-1$ then node $n-1$ exchanges a call with node $f_t^{(n-1)}(x)$.
    [Notice that this can be consistently applied since the size of $\{0, \ldots, n-1\} - \{x\}$ is even and the
    originator x does not need to be called.]

The k–fault–tolerance of the above BROADCAST–SCHEME–ODD($G_{n,k}, x$) is immediate from the correctness of BROADCAST–SCHEME(G_{n-1}, k). $\square$

The above Theorem 4.2 and Corollary 4.1 give the desired bound on $B_k(n)$ when n is odd.

Corollary 4.2 *For each n odd and $k \leq \lfloor 2^{\lceil \log n \rceil} - n + 1 \rfloor$*

$$B_k(n) \leq |E_{n-1,k}| + n - 1 \leq \left(\frac{3}{2}k + L\left(\lceil \frac{n-1}{2^k} \rceil + 1\right) + 3\right)(n-3) + \frac{3}{2}k + 8.$$

5 Fault Tolerant Broadcasting Graphs from Graph Product

The binary hypercube of dimension m, denoted by H_m, is the graph whose nodes are all the binary strings of length m; two nodes are connected by an edge if and only if they differ in exactly one bit.

It is well known that the hypercube is a minimum broadcast graph. Denote by α_i, for $i = 0, \ldots, m-1$, the binary string $\alpha_i = (a_{i,0}, a_{i,1}, \ldots, a_{i,m-1})$ such that

$$a_{i,j} = \begin{cases} 1 & \text{if } j = i \\ 0 & \text{otherwise} \end{cases}.$$

The following is well known to be a broadcasting scheme for the hypercube.

> **HYPERCUBE BROADCASTING**
> **Time unit t, $1 \le t \le m$:** each informed vertex v sends the information to $v \oplus \alpha_{t-1}$, where $\oplus$ denotes the addition modulo 2.

The following property of the above broadcasting scheme will be useful in the sequel.

Property 5.1 *If the originator is the vertex v, then for each node $u = (u_0 \ldots u_{m-1}) \neq v \oplus \alpha_0$ in H_m the paths from v to $u \oplus \alpha_0$, to $u \oplus \alpha_0 \oplus \alpha_1, \ldots$, and to $u \oplus \alpha_0 \oplus \alpha_1 \ldots \oplus \alpha_{m-1}$ are disjoint. For the node $v \oplus \alpha_0$ the paths from v to $v \oplus \alpha_0$, to $v \oplus \alpha_1, \ldots$, and to $v \oplus \alpha_{m-1}$ are disjoint.*

Given graphs $G = (V(G), E(G))$ and $H = (V(H), E(H))$, let us denote by $G[H] = G \times H$ the normal product of G and H, that is, the graph having vertex set $V(G[H]) = V(G) \times V(H)$ and edges set

$$E(G[H]) = \{((u_1, u_2), (v_1, v_2)) \ : \ \text{either } (u_1, v_1) \in E(G) \text{ and } u_2 = v_2 \text{ or } u_1 = v_1 \text{ and } (u_2, v_2) \in E(H)\}$$

Lemma 5.1 *If G is a minimal broadcast graph such that each vertex $v \in V(G)$ has degree $\deg(v) \le \lceil \log |V(G)| \rceil - 1$, then $G[H_k]$ is a minimal k–fault tolerant broadcast graph.*

Proof. Let $V(G) = \{1, \ldots, n\}$. The graph $G[H_k]$ has vertex set $V^* = V(G) \times \{0,1\}^k = \{(v, \alpha) \ : \ v \in V(G), \ \alpha \in \{0,1\}^k\}$ and edge set

$$E^* = \{((i, \alpha), (j, \beta)) \ : \ \text{either } i = j \text{ and } (\alpha, \beta) \in E(H_k) \text{ or } (i, j) \in E(G) \text{ and } \alpha = \beta\}.$$

Notice that $(\alpha, \beta) \in E(H_k)$ if and only if $\alpha = \beta \oplus \alpha_i$ for some unitary string α_i with $0 \le i \le k-1$.

Let the broadcast originator be the node (v_1, β) and suppose that the vertex v_1 during the first time unit of its broadcast in G sends the information to its neighbor v_2. Since the degree of v_1 and

v_2 satisfy $\deg(v_2), \deg(v_1) \le \lceil \log |V(G)| \rceil - 1$, it is clear that there exists a broadcasting strategy in G such that at time unit $\lceil \log |V(G)| \rceil$ the vertices v_1 and v_2 are inactive.

We can then have the following k–fault tolerant broadcasting scheme for the graph $G[H_k]$.

BROADCAST–SCHEME–PRODUCT$(G[H_k])$

(1) **Time unit** t, $(1 \le t \le k)$:
  Each informed vertex (v_1, α) calls the vertex $(v_1, \alpha \oplus \alpha_{t-1})$

(2) **Time unit** $k + t$, $(1 \le t \le \lceil \log n \rceil)$:
  Each informed vertex (v, α), with $v \in V(G)$, calls the node (u, α) if and only if node v should call u at time unit t during the broadcasting from v_1 in G;
  [At time unit $\lceil \log n \rceil$ the vertices v_1 and its first informed neighbor v_2 are inactive]
  At time unit $k + \lceil \log n \rceil$ each vertex (v_i, α), for $i = 1, 2$, calls the vertex $(v_i, \alpha \oplus \alpha_0)$;

(3) **Time unit** $k + \lceil \log n \rceil + t$, $(1 \le t \le k)$:
  Each informed vertex (v, α), with $v \in V(G) - \{v_1, v_2\}$, calls $(v, \alpha \oplus \alpha_t)$;
  if $t < k$, each vertex (v_i, α), for $i = 1, 2$, calls the vertex $(v_i, \alpha \oplus \alpha_{t+1})$
  if $t = k$, each vertex (v_2, α), calls the vertex (v_1, α).

To prove that the above scheme is k–fault tolerant, we need to show that it creates at least $k + 1$ edge disjoint paths from the originator (v_1, β) to each other vertex in V^*. We consider four cases.

Case 1 Let $v \in V(G) - \{v_1, v_2\}$ and $\alpha = (a_0, \ldots, a_{k-1}) \in \{0, 1\}^k$.
The first path is formed by the path

$$(v_1, \beta) \to (v_1, \alpha) \quad \text{created by (1) followed by the path}$$
$$(v_1, \alpha) \to (v, \alpha) \quad \text{created by (2);}$$

The second path is formed by the path

$$(v_1, \beta) \to (v_1, \alpha \oplus \alpha_0) \quad \text{created by (1) followed by the path}$$
$$(v_1, \alpha \oplus \alpha_0) \to (v, \alpha \oplus \alpha_0) \quad \text{created by (2), followed by the call}$$
$$(v, \alpha \oplus \alpha_0) \to (v, \alpha) \quad \text{done by (3) at time unit } k + \lceil \log n \rceil + 1;$$

Let $\alpha^{(\ell-2)} = \alpha_0 \oplus \alpha_1 \oplus \ldots \oplus \alpha_{\ell-2}$, the ℓ-th path, for $3 \le \ell \le k + 1$, is formed by the path

$$(v_1, \beta) \to (v_1, \alpha \oplus \alpha^{(\ell-2)}) \quad \text{created by (1) followed by the path}$$
$$(v_1, \alpha \oplus \alpha^{(\ell-2)}) \to (v, \alpha \oplus \alpha^{(\ell-2)}) \quad \text{created by (2), followed by the path}$$
$$(v, \alpha \oplus \alpha^{(\ell-2)}) \to (v, \alpha \oplus \alpha_{\ell-2}) \quad \text{created by (3) within t.u. } k + \lceil \log n \rceil + \ell - 2, \text{ followed by the call}$$
$$(v, \alpha \oplus \alpha_{\ell-2}) \to (v, \alpha) \quad \text{done by (3) at time unit } k + \lceil \log n \rceil + \ell - 1$$

Case 2 Consider v_2 and $\alpha = (a_0, \ldots, a_{k-1}) \in \{0,1\}^k$.

The first path is formed by the path

$$\begin{array}{rl} (v_1, \beta) \to (v_1, \alpha) & \text{created by (1) followed by the path} \\ (v_1, \alpha) \to (v_2, \alpha) & \text{created by (2);} \end{array}$$

The second path is formed by the path

$$\begin{array}{rl} (v_1, \beta) \to (v_1, \alpha \oplus \alpha_0) & \text{created by (1) followed by the path} \\ (v_1, \alpha \oplus \alpha_0) \to (v_2, \alpha \oplus \alpha_0) & \text{created by (2), followed by the call} \\ (v_2, \alpha \oplus \alpha_0) \to (v_2, \alpha) & \text{done by (3) at time unit } k + \lceil \log n \rceil; \end{array}$$

The ℓ-th path, for $3 \le \ell \le k+1$, is formed by the path

$$\begin{array}{rl} (v_1, \beta) \to (v_1, \alpha \oplus \alpha^{(\ell-2)}) & \text{created by (1) followed by the path} \\ (v_1, \alpha \oplus \alpha^{(\ell-2)}) \to (v_2, \alpha \oplus \alpha^{(\ell-2)}) & \text{created by (2), followed by the path} \\ (v_2, \alpha \oplus \alpha^{(\ell-2)}) \to (v_2, \alpha \oplus \alpha_{\ell-2}) & \text{created by (3) within t.u. } k + \lceil \log n \rceil + \ell - 3, \\ & \text{followed by the call} \\ (v_2, \alpha \oplus \alpha_{\ell-2}) \to (v_2, \alpha) & \text{done by (3) at time unit } k + \lceil \log n \rceil + \ell - 2 \end{array}$$

Case 3 Consider v_1 and $\alpha = (a_0, \ldots, a_{k-1}) \neq \beta + \alpha_0$.

The first path is formed by the path

$$(v_1, \beta) \to (v_1, \alpha) \qquad \text{created by (1).}$$

The second path is formed by the path

$$\begin{array}{rl} (v_1, \beta) \to (v_1, \alpha \oplus \alpha_0) & \text{created by (1) followed by the call} \\ (v_1, \alpha \oplus \alpha_0) \to (v_1, \alpha) & \text{done by (2) at time unit } k + \lceil \log n \rceil \end{array}$$

The ℓ-th path, for $3 \le \ell \le k$, is formed by the path

$$\begin{array}{rl} (v_1, \beta) \to (v_1, \alpha \oplus \alpha^{(\ell-1)}) & \text{created by (1) followed by the path} \\ (v_1, \alpha \oplus \alpha^{(\ell-1)}) \to (v_1, \alpha \oplus \alpha_{\ell-1}) & \text{created by (3) followed by the call} \\ (v_1, \alpha \oplus \alpha_{\ell-1}) \to (v_1, \alpha) & \text{done by (3) at time unit } k + \lceil \log n \rceil + \ell - 2 \end{array}$$

$$\begin{array}{ll} (v_1, \beta) \to (v_2, \beta) & \text{done by (2) at time unit } k+1 \text{ followed by the path} \\ (v_2, \beta) \to (v_2, \alpha) & \text{created by (3) within t.u. } \lceil \log n \rceil + 2k - 1, \\ & \text{followed by the call} \\ (v_2, \alpha) \to (v_1, \alpha) & \text{done by (3) at time unit } \lceil \log n \rceil + 2k \end{array}$$

Case 4 Consider v_1 and $\alpha = (a_0, \ldots, a_{k-1}) = \beta + \alpha_0$.
The first path is formed by the path

$$(v_1, \beta) \to (v_1, \alpha) \quad \text{created by (1)} \tag{19}$$

The ℓ-th path, for $2 \leq \ell \leq k$, is formed by the path

$$\begin{array}{rl} (v_1, \beta) \to (v_1, \beta \oplus \alpha^{(\ell-1)}) & \text{created by (1) followed by the path} \\ (v_1, \beta \oplus \alpha^{(\ell-1)}) \to (v_1, \beta \oplus \alpha_{\ell-1} \oplus \alpha_0) & \text{created by (3) within t.u. } k + \lceil \log n \rceil + \ell - 2, \\ & \text{followed by the call} \\ (v_1, \beta \oplus \alpha_{\ell-1} \oplus \alpha_0) \to (v_1, \alpha) & \text{done by (3) at time unit } k + \lceil \log n \rceil + \ell - 1 \end{array}$$

The $(k+1)$-th path is the same as in Case 3.

By Property 5.1 it follows that for every vertex (v, α) the above mentioned paths are edge disjoint.

□

Let G be the minimum broadcast graph on $n = 2^m - 2$ vertices defined in [19], that is the graph with vertex set $V(G) = \{0, \ldots, 2^m - 3\}$ and edge set

$$E(G) = \{(u, v) \ : \ u + v = 2^r - 1 \bmod 2^m - 2 \text{ for some } 1 \leq r \leq m - 1\}.$$

Theorem 5.1 *The graph $G[H_k]$ is a minimum k–fault tolerant graph.*

Proof. Since each node in G has degree $m - 1 = \lceil \log |V(G)| \rceil - 1$, then by Lemma 5.1 the graph $G[H_k]$ is a minimal k–fault tolerant broadcast graph on $2^k(2^m - 2)$ vertices. Therefore, $B(2^k(2^m - 2)) \leq |E(G[H_k])|$. On the other hand Lemma 2.1 tells that in any minimum broadcast graph on $2^k(2^m - 2) = 2^{m+k} - 2^{k+1}$ vertices the degree of each vertex is lower bounded by $k + (m + k) - (k+1) = m + k - 1$. This implies that $B(2^k(2^m - 2)) \geq (m + k - 1)\frac{n}{2} = |E(G[H_k])|$. □

Theorem 5.2 *For $n = 2^k(2^m - 2^r)$ with $r < m - 1$*

$$\frac{n}{2}(k + m - r) \leq B_k(n) \leq \frac{n}{2}(k + m - \frac{r+1}{2}).$$

Proof. The lower bound follows from Lemma 2.1. To derive the upper bound, let G be the minimal broadcast graph on $n = 2^m - 2^r$, with $r < m - 1$, vertices defined in [19]. This graph satisfies the conditions of lemma 5.1 and has number of edges equal to $(m - \frac{r+1}{2})\frac{n}{2}$. □

6 Minimum 1–Fault Tolerant Graphs

In this section we investigate the minimum number of edges of 1–fault tolerant graphs. In particular we establish the exact value of $B_1(n)$ for each $n \leq 15$ and for $n = 2^m - 6$ with m even.

6.1 Minimum 1–Fault Tolerant Graphs for $n \leq 15$

$\underline{B_1(3) = 3}$: It is immediate to see that the triangle is 1–fault tolerant.

$\underline{B_1(4) = 6}$: See [13].

$\underline{B_1(5) = 7}$: It is easy to see that $B_1(5) \geq 6$. Figure 2 shows the only 3 non–isomorphic graphs on 5 vertices with 6 edges, having minimal local degree 2.

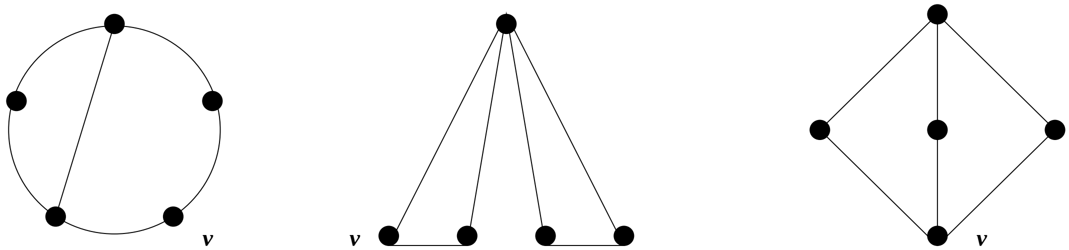


Figure 2:

It is easy to verify that in each case when the originator is the vertex v then the broadcast cannot be done in less than 5 time units. Therefore, $B_1(5) \geq 7$ and this bound is reached by the graph in Figure 3.

$\underline{B_1(6) = 9}$: See [20].

$\underline{B_1(7) = 11}$: It is easy to see that $B_1(7) \geq \lceil \frac{3 \cdot 7}{2} \rceil = 11$, since each vertex must have degree not less than 3. A minimum 1–fault tolerant graph on 7 vertices with 11 edges is shown in Figure 4.

$\underline{B_1(8) = 16}$: See [13].

$\underline{B_1(9) = 13}$:
The following properties for a 1–fault tolerant graph on 9 vertices can be verified.

Property 6.1 *Any vertex v of any minimum 1–fault tolerant graph on 9 vertices with* $\deg(v) =$ 2 *has at least one neighbor u with* $\deg(u) \geq 4$ *which, in turn, has a neighbor of degree at least 3.*

Property 6.2 *In any minimum 1–fault tolerant graph on 9 vertices no two vertices u, v with* $\deg(u) = \deg(v) = 2$ *are adjacent.*

Properties 6.1 and 6.2 implies that $B_1(9) \geq 12$. Moreover, it can be easily seen that there are exactly two non-isomorphic graphs satisfying Properties 6.1 and 6.2 and they are not 1–fault tolerant. Indeed when the originator is the vertex denoted by v in Figure 5 then the broadcast cannot be done in less than 6 time units. Therefore, $B_1(9) \geq 13$. A minimum 1–fault tolerant graph on 9 vertices is shown in Figure 6.

$\underline{B_1(10) = 15}$: See Theorem 6.1 in next section.

$\underline{B_1(11) = 17}$: It is easy to see that $B_1(11) \geq \lceil \frac{3 \cdot 11}{2} \rceil = 17$, since each vertex must have degree not less than 3. A minimum 1–fault tolerant graph on 11 vertices with 17 edges is shown in Figure 7.

$\underline{B_1(12) = 18}$: See Theorem 5.1.

$\underline{B_1(13) = 22}$: Suppose that $G = (V, E)$, with $|E| = B_1(13)$ is a minimum 1-fault tolerant graph. It is easy to see that for every $v \in V$ it must hold $\deg(v) \geq 3$. Moreover, if $\deg(v) = 3$ then v has at least one neighbor u with $\deg(u) \geq 4$. Hence $B_1(13) \geq 21$. Assuming $B_1(13) = 21$ it is easy to see that there exists a vertex u with $\deg(u) = 3$ which has only one neighbor v with $\deg(v) \geq 4$. Let the originator be such a node u. It is clear that the second call from u must be to v. Call u' the node called by u in the first time unit, $\deg(u') = 3$. After 4 time units all 13 vertices must have got the information. Hence, all the edges incident on u' must have been used to inform new vertices. Therefore, it is impossible to construct a second path from the originator u to u'. Hence $B_1(13) \geq 22$. A minimum 1–fault tolerant graph on 13 vertices with 22 edges is shown in Figure 8.

$\underline{B_1(14) = 28}$: See [20].

$\underline{B_1(15) = 30}$: It is easy to see that $B_1(15) \geq \frac{4 \cdot 15}{2} = 30$. A minimum 1–fault tolerant graph on 15 vertices with 30 edges is shown in Figure 9.

6.2 Minimum 1–Fault Tolerant Graphs for $n = 2^m - 6$

In order to describe the desired graphs, we briefly recall the structure and the broadcasting scheme of the minimum broadcast graph $G = (V, E)$ on $n = 2^m - 2$ vertices introduced in [19]. We have $V = \{0, \ldots, 2^m - 3\}$,

$$E = \{(i, j) \ : \ i + j = 2^r - 1 \bmod (2^m - 2) \text{ for some } 1 \leq r \leq m - 1\},$$

and the following possible broadcast schemes.

BROADCAST–SCHEME–a
Time unit t, $(1 \leq t \leq m - 1)$:
  Each informed vertex i calls the vertex j with $i + j = 2^t - 1 \bmod (2^m - 2)$
Time unit m:
  Each informed vertex i calls the vertex j with $i + j = 1 \bmod (2^m - 2)$.

For any $\ell = 1, 2, \ldots, m - 1$ we consider the following sequences

$$X(\ell) = (2^\ell - 1, 2^{\ell+1} - 1, \ldots, 2^{m-1} - 1, 2^1 - 1, \ldots, 2^{\ell-1} - 1, 2^\ell - 1)$$

$$X(\ell) = (2^\ell - 1, 2^{\ell-1} - 1, \ldots, 2^1 - 1, 2^{m-1} - 1, \ldots, 2^{\ell+1} - 1, 2^\ell - 1)$$

and the following schemes:

BROADCAST–SCHEME–b(ℓ)
Time unit t, $(1 \leq t \leq m)$:
  Each informed vertex i calls the vertex j with $i + j = 2^{\ell+t-1} - 1 \bmod (2^m - 2)$

BROADCAST–SCHEME–c(ℓ)
Time unit t, $(1 \leq t \leq \ell)$:
  Each informed vertex i calls the vertex j with $i + j = 2^{\ell-t+1} - 1$
Time unit t, $(\ell + 1 \leq t \leq m)$:
  Each informed vertex i calls the vertex j with $i + j = 2^{\ell-t+m} - 1$

Lemma 6.1 *For any originator $a \in V$ and for any $\ell \in \{1, \ldots, m-1\}$ Broadcast schemes–b(ℓ) and Broadcast schemes–c(ℓ) are correct.*

Proof. We assume in the sequel that all numbers are $\bmod(2^m - 2)$.

Using the Broadcast scheme–b(ℓ) we have

Time unit 1: $a \to 2^\ell - 1 - a$

Time unit 2: $a \to 2^{\ell+1} - 1 - a, \quad 2^\ell - 1 - a \to 2^\ell + a$

Time unit 3: $a \to 2^{\ell+2} - 1 - a, \quad 2^\ell - 1 - a \to 2^{\ell+1} + 2^\ell + a,$

$2^{\ell+1} - 1 - a \to 2^{\ell+1} + a, \quad 2^\ell + a \to 2^{\ell+1} + 2^\ell - 1 - a.$

It is easy to see that after t time units, with $1 \le t \le m-2$ the informed vertices are

$$a,\ 2^{\ell+t-1} - 1 - a$$

and all the vertices

$$b_{\ell+t-2}2^{\ell+t-2} + \ldots + b_\ell 2^\ell + a,\ b_{\ell+t-2}2^{\ell+t-2} + \ldots + b_\ell 2^\ell - 1 - a,$$

with $b_i \in \{0,1\}$ and $(b_\ell, \ldots, b_{\ell+t-2}) \ne (0, \ldots, 0)$.

After $t = m-1$ time units the informed vertices are

$$a, \qquad 2^{\ell-1} - 1 - a \tag{20}$$

and all the vertices

$$b_{m-1}2^{m-1}+\ldots+b_\ell 2^\ell+b_{\ell-2}2^{\ell-2}+\ldots+b_1 2+a, \quad b_{m-1}2^{m-1}+\ldots+b_\ell 2^\ell+b_{\ell-2}2^{\ell-2}+\ldots+b_1 2-1-a \tag{21}$$

with $(b_{m-1}, \ldots, b_1) \ne (0, \ldots, 0)$.

By scheme–b at time unit $t = m$ every informed vertex i calls the vertex j such that $i+j = 2^\ell - 1$. Hence the new informed vertices have either the form

$$j = b_{m-1}2^{m-1} + \ldots + b_\ell 2^\ell + 2^{\ell-1} + b_{\ell-2}2^{\ell-2} + \ldots + b_1 2 + a \tag{22}$$

for all $(b_{m-1}, \ldots, b_1) \ne (1, \ldots, 1)$ or

$$j = b'_{m-1}2^{m-1} + \ldots + b'_\ell 2^\ell + 2^{\ell-1} + b'_{\ell-2}2^{\ell-2} + \ldots + b'_1 2 - 1 - a \tag{23}$$

for all $(b'_{m-1}, \ldots, b'_1) \ne (0, \ldots, 0)$.

It can be easily seen that all the vertices in (20), (21), (22), (23) are different from each other.

The proof for the Broadcast scheme–c(ℓ) is similar. □

Let us consider now any two non adjacent vertices a and b and suppose that the originator is the vertex a.

Lemma 6.2 *There exists $\ell \in \{1, \dots, m-1\}$ such that using the broadcast scheme–b(ℓ) the information originated at a reaches b exactly at the m-th time unit.*

Proof. Without loss of generality, we assume $b = 0$ and $(a, 0) \notin E$, that is, $a \neq 2^k - 1$, $k = 1, \dots, m-1$. By the proof of Lemma 6.1 (formula (22) and (23)) it follows that we have to show the existence of an $\ell \in \{1, \dots, m-1\}$ such that

$$0 = b_{m-1}2^{m-1} + \ldots + b_\ell 2^\ell + 2^{\ell-1} + b_{\ell-2}2^{\ell-2} + \ldots + b_1 2 + a$$

if a is even, $(b_{m-1}, \dots, b_1) \neq (1, \dots, 1)$ and

$$0 = b_{m-1}2^{m-1} + \ldots + b_\ell 2^\ell + 2^{\ell-1} + b_{\ell-2}2^{\ell-2} + \ldots + b_1 2 - 1 - a$$

if a is odd, $(b_{m-1}, \dots, b_\ell, b_{\ell-2}, \dots, b_1) \neq (0, \dots, 0)$.

Let

$$a = x_{m-1}2^{m-1} + \ldots + x_\ell 2^\ell + x_{\ell-1}2^{\ell-1} + x_{\ell-2}2^{\ell-2} + \ldots + x_1 2 + x_0$$

$a \neq 2^k - 1$, for $k = 1, \dots, m-1$, $a \neq 0$.

If $x_0 = 0$ (that is, a is even) it is clear that there exists $r \in \{1, \dots, m-1\}$ for which $x_r = 0$, otherwise $a = 2^{m-1} + \ldots + 2 = 2^m - 2 = 0$. Therefore, we can choose $\ell = i$ for any i such that $x_{i-1} = 0$.

Consider now the case $x_0 = 1$ (that is, a is odd). Since $a \neq 2^k - 1$ for each $k = 1, \dots, m-1$, there exists i for which $x_0 = x_1 = \ldots = x_{i-2} = 1$ and $x_{i-1} = 0$. We choose again $\ell = i$. □

Let $G' = (V', E')$ be the graph on $n = 2^m - 3$ vertices obtained from G by removing the vertex 0 with all its incident edges $(0,1)(0,3), \dots, (0, 2^{m-1} - 1)$. Suppose m is odd. We consider the graph $G^* = (V^*, E^*)$ where $V^* = V'$ and

$$E^* = E' \cup \{(1,3), (7,15), \dots, (2^{m-2} - 1, 2^{m-1} - 1)\}.$$

Property 6.3 *The graph G^* is a minimal broadcast graph on $2^m - 3$ vertices.*

Proof. Let a be the originator. Consider first the case $a \neq 2^k - 1$ for each $k = 1, \dots, m-1$. We apply Lemma 6.2 to find the appropriate ℓ and perform broadcasting from a using the Broadcast scheme–b(ℓ).

If the originator is a vertex $a = 2^{2s+1} - 1$, we perform the broadcasting by scheme–b for $\ell = 2^{2s+2} - 1$. In the $(m-1)$–th time unit the vertex a calls $2^{2s+2} - 1$.

If the originator is a vertex $a = 2^{2s} - 1$, we perform the broadcasting by scheme–c for $\ell = 2^{2s-1} - 1$. In the $(m-1)$–th time unit the vertex a calls $2^{2s-1} - 1$. □

Theorem 6.1 *If m is even then $B_1(2^m - 6) = \frac{(m-1)(2^m-6)}{2}$.*

Proof. From Theorem 2.1 it follows that $B_1(2^m - 6) \geq \frac{(m-1)(2^m-6)}{2}$. On the other hand since each vertex in the graph G^* on $2^{m-1} - 3$ vertices has degree equal to $m - 2$ we can apply Theorem 5.1 to obtain a 1–fault tolerant graph on $2^m - 6$ vertices and $\frac{(m-1)(2^m-6)}{2}$ edges. □

References

[1] R. Ahlswede, H.S. Haroutounian, L.H. Khachatrian, "Messy Broadcasting in Networks", Invited paper in honour of J. Massey on the occasion of his 60th birthday, also Preprint 93-075, SFB 343 "Diskrete Strukturen in der Mathematik", Bielefeld, 1993.

[2] A. Bagchi, and S.L. Hakimi, "Information Dissemination in Distributed Systems with Faulty Units", *IEEE Trans. Comp.* (1994).

[3] A. Bagchi, E.F. Schmeichel, and S.L. Hakimi, "Parallel Algorithms for Information Dissemination by Packets", *SIAM J. on Computing*, 23 (1994), 355-372.

[4] S.C. Chau and A.L. Liestman, Constructing Minimal Broadcast Networks, *J. Comb. Inform. & Systems Sci.*, **10**, 110-122, 1985.

[5] S.-C. Chau and A. L. Liestman, Constructing Fault-Tolerant Minimal Broadcast Networks, *J. Comb. Inf. and. Sci.* 11, 1-18, 1986.

[6] K.A. Berman and M. Hawrycz, "Telephone Problems with Failures", *SIAM J. on Alg. Discr. Meth.*, 7 , 13–17, 1986.

[7] J.–C. Bermond, P. Fraigniaud, and J.G. Peters, "Antepenultimate Broadcasting", *SFU School of Computing Science Technical Report*, CMPT TR 92–03 , 1992.

[8] S. Even and B. Monien, "On the number of Rounds Necessary to Disseminate Information", *Proc. 1st ACM Symp. on Parallel Algorithms and Architectures*, Santa Fe, NM , 318–327, 1989.

[9] A.M. Farley, Minimal Broadcast Networks, *Networks*, **9**, 313–332, 1979.

[10] A. Farley, Reliable Minimum-Time Broadcast Networks, *Proc. Eighteenth SE Conf. on Combinatorics, Graph Theory and Computing* Utilitas Mathematica, 1987.

[11] A.M. Farley, S.T. Hedetniemi, S. Mitchell and A. Proskurowski, Minimum Broadcast Graphs, *Discrete Mathematics*, **25**, 189–193, 1979.

[12] A. Farley and A. Proskurowski, Networks Immune to Isolated Failures, *Networks* 11, 255-268, 1981.

[13] L. Gargano, "Tighter Time Bounds on Fault–Tolerant Broadcasting and Gossiping", *NETWORKS*, 22, 469–486, 1992.

[14] L. Gargano, A. Liestman, J. Peters, and D. Richards, "Reliable Broadcasting", *Disc. Appl. Math.*, 53 (1994), 135–148.

[15] L. Gargano and U. Vaccaro, "On the Construction of Minimal Broadcast Networks", *NETWORKS*, 19, 673–689, 1989.

[16] L. Gargano and U. Vaccaro, "Minimum Time Networks Tolerating a Logaritmic Number of Faults", *SIAM J. on Discrete Mathematics*, 5 , 178–198, 1992.

[17] L. Gargano , A. Rescigno, and U. Vaccaro, "Fault–Tolerant Hypercube Broadcasting via Information Dispersal", *NETWORKS*, **23** ,1993.

[18] M. Grigni and D. Peleg, Tight Bounds on Minimum Broadcast Networks, *SIAM J. Disc. Math*, 1992.

[19] L.H. Khachatrian, H.S. Haroutounian, "Construction of New Classes of Minimal Broadcast Networks", Proc. of the third Int. Colloquium on Coding Theory, 69–77, Armenia, 1990.

[20] L.H. Khachatrian, H.S. Harutounian, "On Optimal Broadcast Graphs", Proc. of the fourth Int. Colloquium on Coding Theory, 65–72, Armenia, 1991.

[21] R.W. Haddad, S. Roy, and A. A. Schäffer, "On Gossiping with Faulty Telephone Lines", *SIAM J. on Alg. Discr. Meth.*, 8 , 439–445, 1987.

[22] S. Hedetniemi, S. Hedetniemi, and A. Liestman, "A Survey of Gossiping and Broadcasting in Communication Networks", *NETWORKS*, 18 , 129–134, 1988.

[23] S.L. Johnsson, C.–T. Ho, "Optimum Broadcasting and Personalized Communication in the Hypercube", *IEEE Transactions on Computer* 38(9), 1249–1268, 1989.

[24] R. Lidl and H. Niederreiter, *Introduction to Finite Fields and their Applications*, Cambridge University Press, 1986.

[25] A. L. Liestman, "Fault-Tolerant Broadcast Graphs", *NETWORKS* 15, 159-171, 1985.

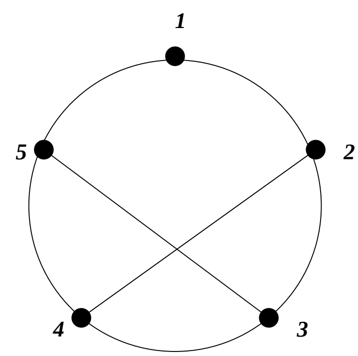
[26] G. Maddaluno, Algorithms for the Constructions of Fault-Tolerant Networks, (in Italian), Thesis, Università di Salerno.

[27] D. Peleg, "A Note on Optimal Time Broadcast in Faulty Hypercubes", *Journal of Parallel and Distributed Computing*, 26, 1995, 132–135.

[28] D. Peleg and A. A. Schäffer, " Time Bounds on Fault-Tolerant Broadcasting", *NETWORKS*, 19, 803-822, 1989.

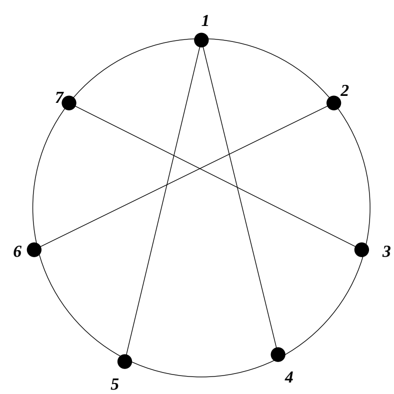
[29] Y. Saad and M.H. Scultz, "Data Communication in Hypercubes", *Parallel Computing* 11, 131–150, 1989.

[30] Q. Stout and B. Wagar, "Intensive Hypercube Communication", *J. Parallel and Distr. Computing*, 10 , 167–181, 1990.



the originator is 1	the originator is 2	the originator is 3
t=1: (1,5)	t=1: (2,1)	t=1: (3,4)
t=2: (1,2),(5,4)	t=2: (2,3),(1,5)	t=2: (3,2),(4,5)
t=3: (2,3)	t=3: (2,4),(3,5)	t=3: (2,1)
t=4: (2,4),(3,5)	t=4: (1,5),(3,4)	t=4: (2,4),(1,5)

Figure 3:



the originator is 1	the originator is 2	the originator is 3	the originator is 4
t=1: (1,5)	t=1: (2,6)	t=1: (3,7)	t=1: (4,1)
t=2: (1,7),(5,6)	t=2: (2,1),(6,5)	t=2: (3,2),(7,1)	t=2: (4,3),(1,2)
t=3: (1,4),(7,3),(6,2)	t=3: (2,3),(1,7),(5,4)	t=3: (1,5),(3,4),(2,6)	t=3: (4,5),(2,6),(3,7)
t=4: (6,7),(4,5),(2,3)	t=4: (1,5),(6,7),(3,4)	t=4: (1,2),(4,5),(6,7)	t=4: (2,3),(5,6),(7,1)

Figure 4:

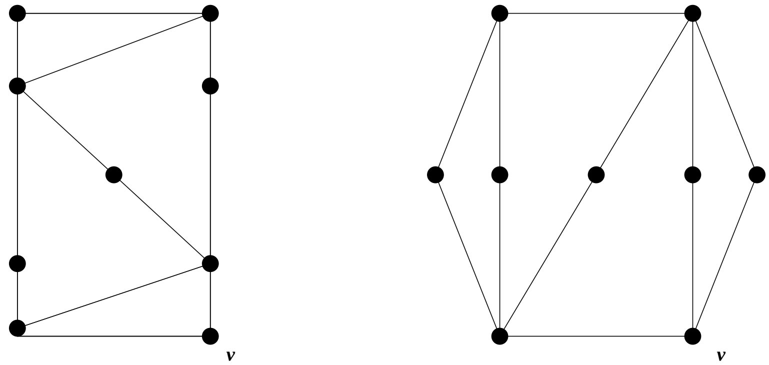
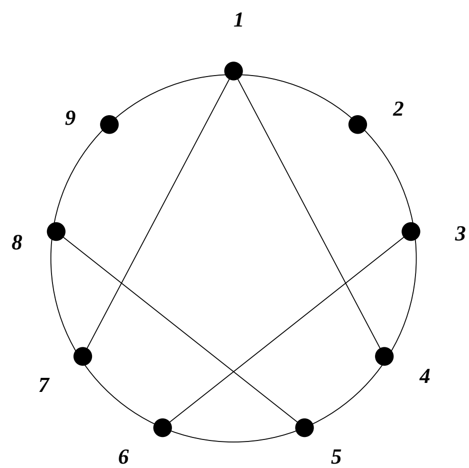


Figure 5:



the originator is 1
t=1: (1,7)
t=2: (1,4),(7,8)
t=3: (1,9),(4,3),(8,5)
t=4: (1,2),(8,9),(3,6)
t=5: (2,3),(4,5),(6,7)

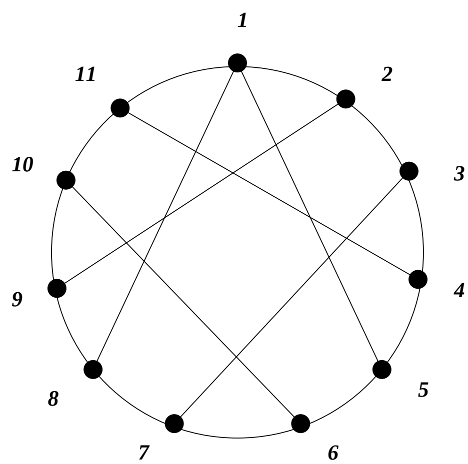
the originator is 2
t=1: (2,3)
t=2: (2,1),(3,6)
t=3: (1,4),(6,7)
t=4: (1,9),(7,8),(4,5)
t=5: (3,4),(5,6),1,7),(8,9)

the originator is 3
t=1: (3,4)
t=2: (3,6),(4,1)
t=3: (3,2),(6,5),(1,7)
t=4: (1,9),(5,8)
t=5: (1,2),(4,5),(6,7),(8,9)

the originator is 4
t=1: (4,1)
t=2: (4,5),(1,7)
t=3: (4,3),(7,6),(1,2),(5,8)
t=4: (8,9)
t=5: (2,3),(5,6),(7,8),(1,9)

the originator is 5
t=1: (5,4)
t=2: (5,6),(4,1)
t=3: (5,8),(6,3),(1,9)
t=4: (1,7),(3,2)
t=5: (1,2),(3,4),(6,7),(8,9)

Figure 6:



the originator is 1
t=1: (1,8)
t=2: (1,5),(8,7)
t=3: (1,2),(5,6),(7,3)
t=4: (1,11),(3,4),(2,9),(6,10)
t=5: (2,3),(4,5),(6,7),(8,9),(10,11)

the originator is 2
t=1: (2,1)
t=2: (2,9),(1,8)
t=3: (2,3),(1,5),(9,10),(8,7)
t=4: (3,4),(7,6),(10,11)
t=5: (1,11),(3,7),(4,5),(6,10),(8,9)

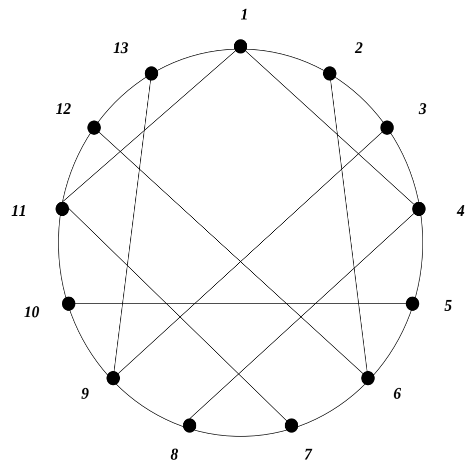
the originator is 3
t=1: (3,2)
t=2: (3,7),(2,1)
t=3: (3,4),(7,8),(1,5),(2,9)
t=4: (1,8),(9,10,),(5,6),(4,11)
t=5: (1,2),(4,5),(6,7),(8,9),(10,11)

the originator is 4
t=1: (4,5)
t=2: (4,3),(5,1)
t=3: (4,11),(3,7),1,8)
t=4: (1,2),(7,6),(8,9),(11,10)
t=5: (1,11),(2,3),(5,6),(7,8),(9,10)

the originator is 5
t=1: (5,6)
t=2: (5,1),(6,7)
t=3: (5,4),(1,11),(7,8)
t=4: (1,2),(4,3),(11,10),(8,9)
t=5: (1,8),(2,9),(3,7),(4,11),(6,10)

the originator is 6
t=1: (6,5)
t=2: (6,7),(5,1)
t=3: (6,10),(7,3),(1,8)
t=4: (1,2),(3,4),(8,9),(10,11)
t=5: (1,11),(2,3),(4,5),(7,8),(9,10)

Figure 7:



the originator is 1
t=1: (1,11)
t=2: (1,4),(11,7)
t=3: (1,13),(11,10),(4,5),(7,8)
t=4: (1,2),(4,3),(5,6),(8,9),(13,12)
t=5: (2,3),(4,8),(5,10),(6,7),(9,13),(11,12)

the originator is 2
t=1: (2,1)
t=2: (2,6),(1,11)
t=3: (1,4),(2,3),(6,12),(11,7)
t=4: (3,9),(4,8),(6,5),(11,10),(12,13)
t=5: (1,13),(3,4),(5,10),(6,7),(8,9),(11,12)

the originator is 3
t=1: (3,4)
t=2: (3,9),(4,1)
t=3: (3,2),(4,8),(1,11),(9,10)
t=4: (2,6),(10,5),(8,7),(9,13,),(11,12)
t=5: (1,2),(4,5),(6,7),(8,9),(10,11),(12,13)

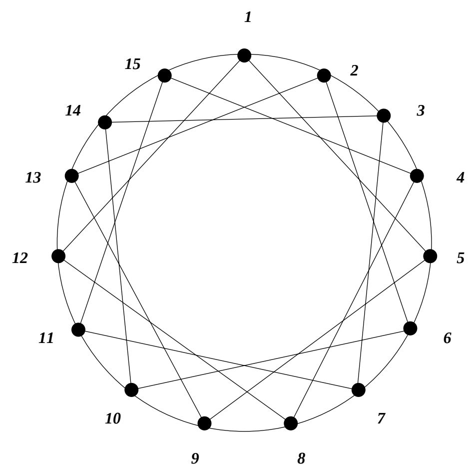
the originator is 4
t=1: (4,5)
t=2: (4,1),(5,6)
t=3: (4,8),(1,11),(6,12)
t=4: (4,3),(1,2),(6,7),(8,9),(11,10),(12,13)
t=5: (1,13),(2,6),(3,9),(5,10),(7,8),(11,12)

the originator is 5
t=1: (5,4)
t=2: (5,6),(4,1)
t=3: (5,10),(4,8),(6,2),(1,11)
t=4: (1,13),(2,3),(6,12),(7,8),(9,10)
t=5: (1,2),(3,4),(6,7),(8,9),(10,11),(12,13)

the originator is 6
t=1: (6,5)
t=2: (6,12),(5,4)
t=3: (6,2),(4,8),(12,11)
t=4: (6,7),(2,3),(4,1),(8,9),(12,13),(11,10)
t=5: (1,2),(3,4),(5,10),(7,8),(9,13),(11,12)

the originator is 7
t=1: (7,11)
t=2: (7,6),(11,1)
t=3: (7,8),(11,10),(6,12),(1,4)
t=4: (4,3),(10,5),(6,2),(8,9),(12,13)
t=5: (1,13),(2,3),(4,8),(5,6),(9,10),(11,12)

Figure 8:



the originator is 1
t=1: (1,5)
t=2: (1,15),(5,9)
t=3: (1,2),(5,6),(15,14),(9,10)
t=4: (1,12),(2,3),(6,7),(8,9),(10,11),(13,14),(15,4)
t=5: (2,6),(3,7),(4,5),(8,12),(9,13),(10,14),(11,15)

Figure 9: